

Claims:

1. A system for the coordination of distributed programs, services and data by using application programs in a network of computers where coordination servers (CoKe) (21,22,23) are running which serve local software systems (LSYS) (18,19,20), where shared objects (9) are used as communication objects to exchange messages and transactions are used to realize communication, said communication objects (9) being uniquely identified by object identification numbers (OID), and only processes possessing a reference to a communication object are granted access to it via the corresponding local coordination server, characterized in that

all coordination servers (21,22,23) together form a global operating system (24),

the local software systems (18,19,20) are at least extended by functions for the control of transactions, for the creation and blocking or non-blocking reading of communication objects, for the specification of transactional predicates and for the creation and supervision of uniquely identified processes granted access to passed communication objects, and

the communication objects (9) are administrated by means of selectable, replication-based distribution strategies, with the application programs not depending on said distribution strategies.

2. A system according to claim 1, characterized in that when choosing the respective distribution strategy, a basic strategy is selected in combination with additional, optional strategy flags.
3. A system according to claims 1 or 2, characterized in that the local software systems are started by the corresponding coordination server (21,22,23).
4. A system according to any one of claims 1 to 3, characterized in that communication objects (9) to which no locally running process possesses a reference any more, are automatically cleared by the corresponding coordination server (21,22,23) or explicitly freed.
5. A system according to any one of claims 1 to 4, characterized in that heterogeneous transactions or subtransactions are distributed to different sites (X,Y,Z) via the coordination servers (21,22,23) which, taken together, behave as a global operating system (24).
6. A system according to any one of claims 1 to 5, characterized in that in the case of updateable objects a transactional read of these objects is provided.
7. A system according to any one of claims 1 to 5, characterized in that the writing into an object, the starting of a subtransaction, the distribution of part of a transaction to another site, the specification of a compensation action or of an on-commitment action are provided as transactional predicate.

8. A system according to any one of claims 1 to 7, characterized in that an on-commitment action is started as a computation if it is sure that a transaction will commit.
9. A system according to any one of claims 1 to 8, characterized in that among the functions for transactions a programmable backtracking of transactional operations, e.g. reading or writing of communication objects (9), is provided to be able to dynamically repair faults or failures in the transactions.

AMENDED CLAIMS:

1. A system for the coordination of distributed programs, services and data by using application programs in a network of computers where coordination servers (CoKe) (21,22,23) are running which serve local software systems (LSYS) (18,19,20), where shared objects (9) are used as communication objects to exchange messages and transactions are used to realize communication, said communication objects (9) being uniquely identified by object identification numbers (OID), and only processes possessing a reference to a communication object are granted access to it via the corresponding local coordination server, with the local software systems (18,19,20) being at least extended by functions for the control of transactions, for the creation, reading and writing of communication objects, and for the creation and supervision of uniquely identified processes, and with the communication objects (9) being administrated by means of replication strategies, characterized in that

all coordination servers (21,22,23) are identical regarding their basic functionality for distributed object, transaction and process management, and taken together, form a global operating system (24),

at least some of the objects are updateable objects, the functions provided for the extension of the local software systems (18,19,20) provide a transactional

blocking read of the updatable object and the processes are granted access to passed communication objects, and distribution strategies are provided for the administration of communication objects (9), with the application programs not depending on said distribution strategies, and which distribution strategies are selectable at least with respect to the recoverability or non-recoverability of communication objects (9) and processes.

2. A system according to claim 1, characterized in that when choosing the respective distribution strategy, a basic strategy is selected in combination with additional, optional strategy flags.

3. A system according to claims 1 or 2, characterized in that the local software systems can be started by the corresponding coordination server (21,22,23).

4. A system according to any one of claims 1 to 3, characterized in that communication objects (9), to which no locally running process possesses a reference any more, are automatically cleared by the corresponding coordination server (21,22,23) or can be explicitly freed.

5. A system according to any one of claims 1 to 4, characterized in that heterogeneous transactions or subtransactions are distributed to different sites (X,Y,Z) via the coordination servers (21,22,23) which, taken together, behave as a global operating system (24).

6. A system according to any one of claims 1 to 5, characterized in that a non-blocking transactional read is provided for updateable objects.
7. A system according to any one of claims 1 to 5, characterized in that the writing into an object, the starting of a subtransaction, the distribution of part of a transaction to another site, the specification of a compensation action or of an on-commitment action are provided as transactional predicate.
8. A system according to any one of claims 1 to 7, characterized in that, an on-commitment action is started as a computation if it is sure that a transaction will commit.
9. A system according to any one of claims 1 to 8, characterized in that among the functions for transactions a programmable backtracking of transactional operations, e.g. reading or writing of communication objects (9), is provided to be able to dynamically repair faults or failures in the transactions.

Add
A₁